Amendments to the Specification:

Please replace the paragraph beginning at page 3, line 2, with the following rewritten paragraph:

The merged SP/PE0 building block unit logically functions as a single context controller and by virtue of the merged PE provides supporting interfaces that allow additional PEs to be attached. In this single context controller environment, the ~~S/P~~SP/PE-bit is used to determine whether an instruction is to be executed in the SP only or is to be executed in the PE array. In one aspect of the present invention, the ~~S/P~~SP/PE-bit is used in a 1x1 array core to determine which register file, the SP's or the PE's, is to be accessed for each instruction execution. By treating the ~~S/P~~SP/PE-bit as a context-0/context-1 bit, the selection between two different register spaces effectively doubles the size of the register space for the SP. Thus, the 1x1 array core can be viewed as a single processor containing two register contexts that share a common set of execution units.

Please replace the paragraph beginning at page 3, line 11, with the following rewritten paragraph:

Note that this approach of using the ~~S/P~~SP/PE-bit for context switching purposes requires that for an instruction to access the PE register space, it must set the SP/PE bit in the instruction word to indicate it is a PE instruction. The implication of this requirement is that different forms of instructions are required to be used for accessing different registers. If it is desired to make use of both register files in a 1x1, for different contexts for example, the code must be explicitly targeted by using either PE or SP instructions. This limitation does not allow for seamless

context switching between tasks since the task code is not uniform. As addressed further below, the present invention advantageously addresses these and other limitations providing improved context switch control.

Please replace the paragraph beginning at page 3, line 20, with the following rewritten paragraph:

Multiple register contexts are obtained in the ManArray processor by controlling how the array ~~S/P~~SP/PE-bit in the ManArray instruction format is used in conjunction with a context switch bit (CSB) for the context selection of the PE register file or the SP register file. In arrays consisting of more than a single PE, the software controllable context switch mechanism is used to reconfigure the array to take advantage of the multiple context support the merged SP/PE provides. For example, a 1x1 can be configured as a 1x1 with context-0 and as a 1x0 with context-1, a 1x2 can be configured as a 1x2 with context-0 and as a 1x1 with context-1, and a 1x5 can be configured as a 1x5 with context-0 and as a 2x2 with context-1. Other array configurations are clearly possible using the present invention. In the 1x5/2x2 case, the two contexts could be a 1x5 with the sequential control context in the SP register files with context-0 and a 2x2 array context, where the sequential control context uses the PE0's register files with context-1.

Please replace the paragraph beginning at page 4, line 16, with the following rewritten paragraph:

Fig. 2 provides a high-level view of the basic function of the ~~S/P~~SP/PE-bit and context switch bit (CSB) for improved context switch control in accordance with the present invention;

Please replace the paragraph beginning at page 4, line 18, with the following rewritten paragraph:

Fig. 3 specifies the logical operation of various array configurations for different settings of the CSB and the instruction's ~~S/P~~SP/PE-bit;

Please replace the paragraph beginning at page 9, line 18, with the following rewritten paragraph:

To provide for efficient context switching within a ManArray processor, a processor mode bit is provided in a control register such as a processor state register in a miscellaneous register file (MRF). This bit is identified as a context switch bit (CSB). Fig. 2 illustrates a functional view of a system 200 for implementing the present invention. An ~~S/P~~SP/PE-bit and CSB bit control logic unit 202 contains the CSB and override logic. The control logic unit 202 provides enable signals 204 and 206 to multiplexers 208 and 210, respectively, to select where the result data from the execution units 212 are to be written. The result data is selectably written either to the SP reconfigurable compute register file 214 or to the PE reconfigurable compute register file 216. The control logic unit 202 also provides a select signal 218 to a multiplexer 220 to control which block of registers 214 or 216 that execution units 212 read data from. It is noted that in Fig. 2, the execution units 212 in the ManArray iVLIW processor may advantageously comprise five heterogeneous execution units which correspond to the five execution units 131 in Fig. 1. Also, the buses, multiplexers, and select control signals shown in Fig. 2 are indicated with multiple lines since in the ManArray processor such as shown in Fig. 1 there are eight 32-bit read ports and four 32-bit write ports for each 16x32-bit portion of both of the reconfigurable

register files and each requires separate selection and control depending upon the instruction in execution and the machine state.

Please replace the paragraph beginning at page 10, line 11, with the following rewritten paragraph:

Specifically, the CSB bit in conjunction with the ~~S/P~~SP/PE-bit in PE0's control logic allows efficient context switching between tasks. Control specification 300 of Fig. 3 lists three exemplary array configurations and describes the register file use and array operating configuration for SP or PE instructions, as specified by the instruction's ~~S/P~~SP/PE-bit, depending upon the setting of the CSB bit. Table 310 indicates the ManArray architecture definition of the ~~S/P~~SP/PE-bit, which is present in the execution units' instruction formats. In general, other register files including the reconfigurable compute register files are shared between contexts. Specifically, in Fig. 3, the register files that are indicated to be shared are the address register file (ARF), the compute register file (CRF), and selected MRF and special purpose registers (SPRs) used by the execution units. The physical MxN column 304 indicates the physical array organization of PEs in the core processor, while the operating MxN column 312 depends upon the CSB value. It is noted that with the CSB bit set to zero, as seen in control specification entries 320, 322, 330, 332, 340, and 342, the SP operates in context-0 with SP instructions only executing in the SP on SP resources and PE instructions only executing in any or all of the PEs on PE resources. With the CSB bit set to a one, as seen in control specification entries 324, 326, 334, 336, 344, and 346, the SP operates in context-1 which uses the PE0's register files. As described by this invention, each MxN core is a two context processor where one of the contexts

uses SP-only resources for sequential control while the other context uses PE0's resources for sequential control.

Please replace the paragraph beginning at page 11, line 6, with the following rewritten paragraph:

By controlling the CSB-bit, an operating system (OS) can select a "context" for a task. In the 1x1 case, entries 320-326, where no PE instructions are used in a program, the core processor acts as a 1x0 with two contexts that the OS can freely assign as required by an application. In this 1x1 case, use of the CSB bit, rather than dependence on the ~~S/P~~SP/PE-bit only, allows the task code to be written in a uniform manner when using only the SP forms of instructions. Using PE instructions on PE0 even when the CSB bit is set to a 1, entry 326, is not likely an advantage, but can be optionally allowed effectively sharing PE0's context-1 register files between the SP and PE0.

Please replace the paragraph beginning at page 18, line 3, with the following rewritten paragraph:

The ManArray core indirect VLIW processor consists of an array controller sequence processor (SP) merged with a processing element (PE0) closely coupling the SP with the PE array and providing the capability to share execution units between the SP and PE0. Consequently, in the merged SP/PE0 a single set of execution units are coupled with two independent register files. To make efficient use of the SP and PE resources, the ManArray architecture specifies a bit in the instruction format, the ~~S/P~~SP/PE-bit, to differentiate SP instructions from PE instructions. Multiple register contexts are obtained in the ManArray

processor by controlling how the array ~~S/P~~SP/PE-bit in the ManArray instruction format is used in conjunction with a context switch bit (CSB) for the context selection of the PE register file or the SP register file. In arrays consisting of more than a single PE, the software controllable context switch mechanism is used to reconfigure the array to take advantage of the multiple context support the merged SP/PE provides. For example, a 1x1 can be configured as a 1x1 with context-0 and as a 1x0 with context-1, a 1x2 can be configured as a 1x2 with context-0 and as a 1x1 with context-1, and a 1x5 can be configured as a 1x5 with context-0 and as a 2x2 with context-1. Other array configurations are clearly possible using the present techniques. In the 1x5/2x2 case, the two contexts could be a 1x5 array (context-0) and a 2x2 array (context-1).